

Title: Fast Randomized Point Location without Preprocessing
in Two- and Three-Dimensional Delaunay Triangulations

Author(s): Ernst P. Mucke
Isaac A. Saias
Binhai Zhu

Submitted to: *Computational Geometry Theory and Applications*

Los Alamos
National Laboratory

Los Alamos National Laboratory, an affirmative action/equal opportunity employer, is operated by the University of California for the U.S. Department of Energy under contract W-7405-ENG-36. By acceptance of this article, the publisher recognizes that the U.S. Government retains a nonexclusive, royalty-free license to publish or reproduce the published form of this contribution, or to allow others to do so, for U.S. Government purposes. Los Alamos National Laboratory requests that the publisher identify this article as work performed under the auspices of the U.S. Department of Energy. The Los Alamos National Laboratory strongly supports academic freedom and a researcher's right to publish; as an institution, however, the Laboratory does not endorse the viewpoint of a publication or guarantee its technical correctness.

Fast Randomized Point Location Without Preprocessing in Two- and Three-dimensional Delaunay Triangulations

Ernst P. Mücke

Isaac Saias

Binhai Zhu

Los Alamos National Laboratory, Los Alamos, New Mexico *

Abstract

This paper studies the point location problem in Delaunay triangulations without preprocessing and additional storage. The proposed procedure finds the query point by simply “walking through” the triangulation, after selecting a “good starting point” by random sampling. The analysis generalizes and extends a recent result for $d = 2$ dimensions by proving this procedure to take expected time close to $O(n^{1/(d+1)})$ for point location in Delaunay triangulations of n random points in $d = 3$ dimensions. Empirical results in both two and three dimensions show that this procedure is efficient in practice.

1 Introduction

Point location is one of the classical problems in computational geometry and has various applications of practical relevance, for example, in the areas of geographic information systems (GIS) or computer-aided design and engineering (CAD/CAE). The problem is well studied in the computational geometry literature and several theoretically optimal algorithms have been proposed. Unfortunately, algorithms that are optimal in theory do not necessarily yield to good practical performance. This is also true in the case of point location, mainly because of the necessary preprocessing time and additional storage requirements.

Actual engineering implementations often use tree structures to guide the point location, for example, the “alternating digital tree” of Bonet and Péraire [BP91], which typically come very close to the theoretically optimal logarithmic time complexity of the problem. However, all these methods require a certain amount of preprocessing for the creation of additional data structures (and their maintenance, for the dynamic version of the problem). “Bucketing” algorithms, for example, Asano et al. [AEI⁺85], can even achieve $O(1)$ search time, on average, for uniform distribution, for input in a bounded domain, but they, too, require extra preprocessing, especially within each bucket, and additional storage. Here, we will discuss a technique that is efficient in

*The Authors may be contacted at: E. P. Mücke, epm@ansys.com, ANSYS, Inc., 201 Johnson Road, Houston, PA 15342-1300, USA. I. Saias, isaac@lanl.gov, and B. Zhu, bhz@lanl.gov, both at Los Alamos National Laboratory, CIC-19 Computer Research & Applications Group, M.S. B256, Los Alamos, NM 87545, USA.

practice, uses *no* preprocessing time, *no* additional storage, and, as a bonus, could not be easier to implement.

The point location problem in its full generality deals with locating query points in arbitrary subdivisions. This work, however, focuses on point location in triangulations (in fact, the analysis is even further restricted to Delaunay triangulations of random points). This is justified because regions of arbitrary subdivisions can be triangulated; moreover, the query problem in triangulations itself occurs quite frequently in practice, for example, in mesh generation and finite-element analysis (FEA).

Point location by walking. The basic idea is straightforward and not at all new; it goes back to early work on constructing Delaunay triangulations in 2D and 3D [GS78, B81]. Given a Delaunay triangulation \mathcal{D} of a set X of n points in \mathbb{R}^d , and a target point q ; in order to locate the simplex (triangle in 2D, tetrahedron in 3D) in \mathcal{D} containing q , start at some arbitrary simplex in \mathcal{D} and then “walk” from simplex to neighboring simplex “in the general direction” of the target point q . The underlying assumption is that the \mathcal{D} is given by an internal representation allowing constant-time access between neighboring simplices. The list of suitable data structures includes the 2D quad-edge data structure [GS85], the edge-facet structure in 3D [DL89], its specialization and compactification to the domain of 3D triangulations [M93], or its generalization to d dimensions [B93].

The pseudo-code for the simple “walking” method in 2D can be found in [GS85]. Note that this procedure is only guaranteed to terminate in Delaunay triangulations. For arbitrary triangulations it might go into an infinite loop; see, for example, [GH95]. However, the infinite loop can be broken (for all practical purposes) by introducing randomness when selecting which neighboring simplex to visit next; see section 6.

The “walking” method has been ignored by most theoreticians in computational geometry since not much can be said about its performance theoretically, other than it is “expected” to take time proportional to $n^{1/d}$ when the points are randomly distributed [GS78, B81]. However, because of its exceptional simplicity, the method is used by practitioners in the geometric computing community, in particular, in mesh generation for FEA [GH92, BL95].

The jump-and-march. In the following, we modify the method somewhat. First, we “march” towards q by strictly traversing the simplices intersected by a line segment L , starting at a vertex of the initial simplex, and ending in q . This makes it easier to analyze the procedure (and trivially eliminates the above mentioned infinite-loop problem); although, as discussed empirically in section 6, it seems to reduce the expected number of visited simplices only marginally, and actually degrades the actual performance. Second, we “jump” to a good starting point via random sampling on the point set $\{X_1, X_2, \dots, X_n\}$. This enhances the overall procedure significantly; in a sense, we are simulating the effects of the bucketing approach by random sampling.

Given the Delaunay triangulation \mathcal{D} of these n points $\{X_1, X_2, \dots, X_n\}$, and a query point q , the following procedure locates the simplex of \mathcal{D} containing q , if such a simplex exists.

- (1) Select m points Y_1, \dots, Y_m at random and without replacement from X_1, \dots, X_n .
- (2) Determine the index $j \in \{1, \dots, m\}$ minimizing the distance $d(Y_j, q)$.
Set $Y = Y_j$.

- (3) Locate the simplex containing q by traversing all simplices intersected by the line segment (Y, q) .

Step (3), that is, the straight “march,” can be implemented in constant time per simplex visited, once the initial simplex, intersected by L and incident to “starting point” Y , is determined.

Motivated by the positive empirical results of [M93], where the jump-and-march (or rather, the jump-and-*walk*; see also Section 6) is used to implement the randomized incremental flip algorithm to construct 3D Delaunay triangulations, this procedure was recently analyzed for the 2D case, with the result that the expected query time is $O(n^{1/3})$ when the points are randomly distributed [DMZ95]. This result, in turn, builds on the work of Bose and Devroye [BD95] who prove that for any line segment L the expected number of intersected triangles is proportional to $|L|n^{1/2}$.

In the following, both results are extended to \mathbb{R}^3 , showing that the jump-and-march point location in spatial Delaunay triangulations of n random points has an expected running time of $O(\delta(n)^{1/4} n^{1/4} (\log n / \log \log n)^{3/4})$, where $\delta(n)$ denotes the *expected* degree of a Delaunay vertex. A result by Bern et al. [BEY91] on the expected *maximum* degree would give $\delta(n) = O(\log n / \log \log n)$. On the other hand, Dwyer [Dwy91] shows that $\delta(n) = O(1)$ for any fixed dimension d , under the assumption that the points are chosen uniformly at random in a d -dimensional ball. In any case, it is always a fair assumption that Delaunay triangulations occurring in problems of practical relevance are only of linear size (rather than worst-case quadratic size), and we can immediately argue that $\delta(n)$ is constant for all practical purposes, yielding an expected running time close to $O(n^{1/4})$. This compares well to the theoretically optimal $O(\log n)$ bound, at least for practical sizes of input data; for example, $n^{1/4} / \log_2 n < 2.5$, for n in the range up to 10^7 .

On a theoretical side, our work addresses and solves two difficult issues. First, when proving “probabilistic impossibility results” for Delaunay triangulations one is naturally led to define volumes and to argue that these volumes are likely to contain some Delaunay vertices. One must be careful though to define (as much as possible) these volumes *independently* from the vertices. We achieve this difficult task in 3D. Second, the perturbing effect of the boundary is very well-known. The probabilistic model of [BEY91], for instance, was designed to analyze typical properties of Delaunay triangulations away from the boundary. Here, we provide a specific estimate of the range of this perturbation. Our methods seem well suited to bring even more precise results.

Outline. The paper is organized as follows. In Sections 2 and 4, we first generalize the result of [BD95] regarding the intersection of a line segment with a random Delaunay triangulation to 3D. Then, we generalize the proof of [DMZ95] to 3D. Section 3 presents an outline of the proof. In Sections 5 and 6, we present empirical results over randomly generated point sets ranging from $n = 1000$ to 50000. Our tests confirm that the method is efficient in practice and compares with theoretically optimal $O(\log n)$ methods, at least in the above range, which seems to be of most relevance for practitioners in GIS and CAD.

2 Statement of Results

Let C be a convex domain of \mathbb{R}^3 and let α and β be two reals such that $0 < \alpha < \beta$. We say that a probability measure P is an (α, β) -measure over C if $P[C] = 1$ and if we have $\alpha \lambda(S) \leq P[S] \leq \beta \lambda(S)$ for every measurable subset S of C , where λ is the usual Lebesgue measure.¹ An \mathbb{R}^3 -valued random variable X is called an (α, β) -random variable over C if its probability law $\mathcal{L}(X)$ is an (α, β) -measure over C . A particular and important example of an (α, β) -measure P is when P is a probability measure with density $f(x)$ such that $\alpha \leq f(x) \leq \beta$ for all $x \in C$. One of the advantages of our more general notion is that it allows for a probability measure charging only points with rational coordinates: this is the case for most computer simulations. This probabilistic model was introduced in [BD95]. The Poisson model of [BEY91] is related to ours in the sense that, conditioned on the number n of points observed over a finite volume, the probability distribution is uniform, that is, an (α, α) -measure.

Below is our main result on the expected running time of the jump-and-march algorithm, when applied on \mathcal{D} , the Delaunay triangulation of n random points in \mathbb{R}^3 .

Theorem 1. Let C be a bounded convex set of \mathbb{R}^3 having small curvature. Let X_1, \dots, X_n be n points drawn independently in C from an (α, β) -measure. Then there exist constants c_1, c_2 and c_3 depending only upon α, β and C such that the following holds. Assume that $m \geq n^{1/5}$ and that the query point is selected independently of X_1, \dots, X_n and is at distance of at least $c_1/n^{1/18}$ from the boundary ∂C . Then the expected time of the jump-and-march algorithm is bounded by

$$c_2 m \delta(n) + c_3 (n/m)^{1/3} \log n / \log \log n,$$

where $\delta(n)$ is the expected vertex degree of the Delaunay triangulation. In particular, the expected time is optimized to $O(\delta(n)^{1/4} n^{1/4} (\log n / \log \log n)^{3/4})$ with the choice of $m = \Theta(n^{1/4} / \delta(n)^{3/4} (\log n / \log \log n)^{3/4})$.

The proof of Theorem 1 rests on the following theorem.

Theorem 2. Let C be a bounded convex set of \mathbb{R}^3 having small curvature. Let X_1, \dots, X_n be n points drawn independently in C from an (α, β) -measure. Then there exist constants c_4 and c_5 depending only upon α, β and C such that the following holds. Let L be a segment in C being at distance of at least $c_4(\log n/n)^{1/3}$ from the boundary ∂C . Let N be the number of intersections between L and \mathcal{D} . Then:

$$E[N] \leq c_5 |L| n^{1/3} \log n / \log \log n.$$

We can easily extend Theorem 2 to the case where L is a random segment *independent* of the n points X_1, \dots, X_n . For this, define the event $B = \{d(L, \partial C) \geq c_4(\log n/n)^{1/3}\}$. We then have $E[N|B] \leq c_5 E[|L| | B] n^{1/3} \log n / \log \log n$. In Section 4, we first prove Theorem 2 following the same ideas as [BD95]; however, we would like to point out that the technical details are quite different in 3D and more difficult. Given Theorem 2, it is easy to generalize the result of [DMZ95] to obtain Theorem 1.

¹Note that the relation $\lambda(C) \leq 1/\alpha < \infty$ implies that C has finite area. The convexity of C then implies that C is bounded (that is, that C is included in some finite ball.)

Theorems 1 and 2 are true without the requirement that C have a small curvature. We place this requirement to emphasize that the constants intervening in the theorems are big unless the curvature of C is small.

To finish let us clarify some rather difficult probabilistic issues underlying our results. It is important to realize that our probability results involve *two* very different sources of randomness. The first source of randomness is the (α, β) -measure generating the random input X_1, \dots, X_n . In particular, the expected vertex degree $\delta(n)$ is computed with respect to this probability measure. The second source of randomness is the randomness introduced in step (1) of our randomized algorithm. The expected values presented in Theorems 1 and 2 are computed over both sources of randomness. In addition, one should realize that, in Theorem 1, the query point is *not* sampled from any fixed random distribution: this point is chosen non-deterministically. One way to express that, is to say that the query point is selected by an adversary. The adversary can itself use randomness and can be assumed to know the two probability distributions used in our analysis; of course, a critical restriction is that the adversary must select the query point unaware of the realization of these distributions.

3 General Outline of the Proof

As mentioned above, Theorem 1 is a rather simple consequence of Theorem 2. We outline first the proof of Theorem 1. In general, the time required to walk through a segment L is proportional to the number of Delaunay tetrahedra crossed by L . By Theorem 2, this number is close to $|L|n^{1/3}$ provided that the segment L is chosen *independently* of the triangulation \mathcal{D} . To apply this result to the walk-phase, when one walks from Y (the point selected in the jump-phase) to q (the query point), we proceed as follows. We consider the triangulation \mathcal{D}' induced by the $n - m$ points not used in the jump-phase. One can easily show that the total time required by the walk-phase is bounded by the time to walk from Y to q in \mathcal{D}' , plus the total tetrahedron-degree (in \mathcal{D}) of the m points Y_1, \dots, Y_m . The point Y depends only on Y_1, \dots, Y_m and is independent from the triangulation \mathcal{D}' . One can therefore apply Theorem 2 and show that the expected time to walk from Y to q in \mathcal{D}' is $O(\frac{1}{m^{1/3}}(n - m)^{1/3} \log n / \log \log n)$. (The assumption $m \geq n^{1/5}$ is used there.) On the other hand, the expected tetrahedron-degree (in \mathcal{D}) of each point Y_1, \dots, Y_m is equal to $\delta(n)$. As mentioned in Section 1, $\delta(n) = O(1)$ in practical situations and always at most $O(\log n / \log \log n)$. This establishes Theorem 1.

We now turn to an outline of the proof of Theorem 2. This proof is longer and requires several technical lemma. To bound the (expected) number of tetrahedra traversed by a segment L , we cover L with little balls of appropriate radius and reduce the problem to bound the (expected) number N_2 of tetrahedra intersected by each of these balls $\mathcal{B}(y, r)$. (This idea is directly inspired by [BD95].) To estimate (the expected value of) N_2 we introduce and analyze N_1 : N_1 is the number of sites X_i such that one of the Delaunay tetrahedra incident to X_i intersects $\mathcal{B}(y, r)$. To connect N_2 to N_1 we use an extension of a result established in [BEY91]: with high probability, for every site X_i , the number of Delaunay tetrahedra incident to X_i is $O(\log n / \log \log n)$. This allows us to derive that $E[N_2] = O(E[N_1] \log n / \log \log n)$.

We have reduced the original problem to the one of estimating an upper bound of

$E[N_1]$. For this, we consider a given site X_1 and estimate the probability that one of the Delaunay tetrahedra incident to X_1 intersects $\mathcal{B}(y, r)$: $E[N_1]$ is equal to n times this probability. Our majoration hinges on a geometric argument: we consider a set of three little balls A_1, A_2, A_3 positioned at equal distance between X_1 and y . (This set depends upon X_1 and y but not upon X_2, \dots, X_n .) We show that every ball containing X_1 and intersecting $\mathcal{B}(y, r)$ must contain one of the three balls A_1, A_2, A_3 . By the Delaunay empty-ball condition, no Delaunay tetrahedron incident to X_1 can intersect $\mathcal{B}(y, r)$ if each of the three balls A_1, A_2, A_3 contains a site X_j . This fact allows us to bound the probability that one of the Delaunay tetrahedra incident to X_1 intersects $\mathcal{B}(y, r)$. This in turn allows us to bound $E[N_1]$, then $E[N_2]$ and to prove Theorem 2.

To finish we mention some thorny technical issues encountered in the proof. The main customary argument used when proving probabilistic upper bounds for Delaunay triangulations consists of defining volumes, (usually balls), and to argue that no Delaunay site should be contained in these volumes. One must be careful to define as much as possible these volumes independently of the Delaunay triangulation. To achieve that purpose we introduce in Section 4.1 the notion of a “spindle”: the spindle is composed of the three balls A_1, A_2, A_3 . The independence of the spindle from the Delaunay triangulation is used critically in the proof of Lemma 5.

A second difficulty comes from the fact that all three balls A_1, A_2, A_3 might not be fully contained in C when X_1 is too close to the boundary of C . The formal treatment of this situation complicates the estimation of $E[N_1]$ done in Lemma 5.

4 Probabilistic Analysis

A face F (that is, an edge, triangle, or tetrahedron) of the Delaunay triangulation \mathcal{D} is called a *Delaunay face*; we denote this by $F \in \mathcal{D}$. For every four points x, y, t, z in general position we let $\mathcal{B}(xytz)$ be the unique ball circumscribed to these four points. Also, for every point x and $r \geq 0$, we let $\mathcal{B}(x, r)$ denote the ball of radius r centered at point x .

Our arguments proceed as follows. First, we state and prove a geometric lemma (Lemma 3), which will later be used in the proof of Lemma 5. Second, we give a set of probabilistic lemmas (Lemma 4–7), which, in a third step, we combine to prove Theorems 1 and 2.

4.1 A Geometric Lemma

Let l be a positive number. We define an *l-spindle* to be a geometric object composed of an axle surrounded by three balls:

- The axle is a line segment of length l . We let xy denote that line segment.
- At $q = \frac{x+y}{2}$, we attach three spokes of length $k_1 l$. The spokes are in a plane perpendicular to the axle; each pair of spokes forms an angle of $\frac{2\pi}{3}$.

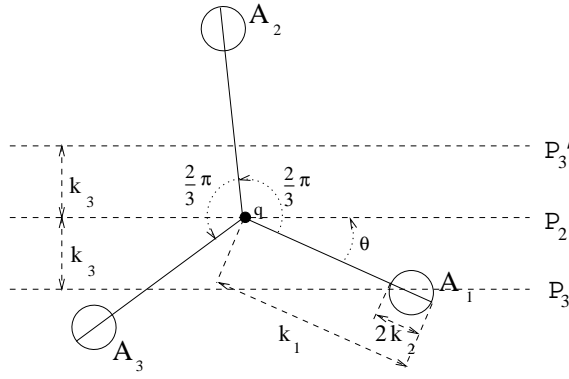


Figure 1: Defining k_1, k_2 and k_3 .

- We place a ball A_i at the end of each of the three spokes $i = 1, 2, 3$. These balls have radius $k_2 l$ and their centers are at distance $(k_1 - k_2)l$ from q .

The collection of the three balls A_1, A_2, A_3 is called the *wheel* of the spindle. For Lemma 3 below, we restrict the numbers k_1, k_2, k_3 to be positive such that $k_1 + k_3 < \frac{1}{2}$ and $3k_2 + 2k_3 < k_1$. These numbers are considered constants throughout this text. Nevertheless, we will impose in the proof of Lemma 5 some additional restrictions on the choice of these “constants”.

Figure 1 illustrates the situation. The plane of the figure (which is the plane of the spindle which contains the centers of A_1, A_2, A_3) is perpendicular to xy and goes through q . Let \mathcal{P}_2 be any plane going through q and containing xy , and let \mathcal{P}_3 and \mathcal{P}_3' be the two planes parallel to \mathcal{P}_2 and at distance k_3 of \mathcal{P}_2 .

Let also θ denote the angle between \mathcal{P}_2 and the spoke of A_1 , and consider the case where $\theta = \frac{\pi}{6}$. In this case, the balls A_1 and A_3 are at the same distance of \mathcal{P}_2 , namely: $d(A_1, \mathcal{P}_2) = d(A_3, \mathcal{P}_2) \stackrel{\text{def}}{=} d_{\min}$. As illustrated in Figure 2, $d_{\min} = l_1 \sin(\frac{\pi}{6}) = \frac{l_1}{2}$. The quantities l_1, l_2 and l_3 satisfy the three linear relations $k_1 = l_1 + l_2 + l_3$ and $l_3 = k_2 = l_2 \sin(\frac{\pi}{6})$ allowing us to conclude that $d_{\min} = \frac{l_1}{2} = \frac{1}{2}(k_1 - 3k_2)$.

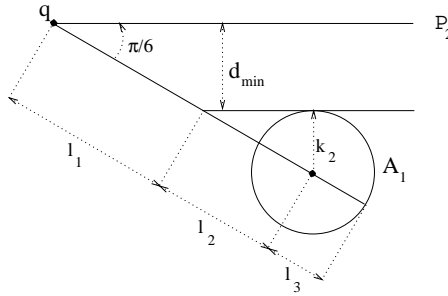


Figure 2: The case where $\theta = \pi/6$.

For a general value of θ , we can easily check that there is (at least) one ball A_i such that $d_{\min} \leq d(A_i, \mathcal{P}_2)$ and such that A_i is on same side of \mathcal{P}_2 as \mathcal{P}_3 . Symmetrically,

there is (at least) one ball A_j with $d_{\min} \leq d(A_j, \mathcal{P}_2)$ and which is on the same side of \mathcal{P}_2 as \mathcal{P}_3' . Therefore, the choice $k_3 < d_{\min}$ ensures that, for every value of θ , there are two balls A_i and A_j such that A_i is “below” \mathcal{P}_3 and such that A_j is “above” \mathcal{P}_3' . (The terms “below” and “above” are used for simplification and are intuitively made clear by Figure 1.)

Lemma 3. Let x, y, x_1, \dots, x_n be points in \mathbb{R}^3 with $l = d(x, y)$ and let k_1, k_2, k_3 be three positive numbers such that $k_1 + k_3 < \frac{1}{2}$ and $3k_2 + 2k_3 < k_1$. Let \mathcal{S} be any l -spindle whose axle is xy and let as before A_1, A_2, A_3 be the three balls of its wheel. Finally, let \mathcal{D} be the Delaunay triangulation of x, x_1, \dots, x_n . If all three balls A_1, A_2, A_3 each contain a point x_j , then there exists no tetrahedron in \mathcal{D} incident to x which intersects (touches, crosses or contains) the ball $\mathcal{B}(y, k_3l)$.

Proof: We begin by noting that $k_1 + k_3 < \frac{1}{2}$ implies that the balls $\mathcal{B}(q, k_1l)$ and $\mathcal{B}(y, k_3l)$ do not intersect (see also Figure 3). We are going to show that every ball \mathcal{B} having x on its boundary and intersecting ball $\mathcal{B}(y, k_3l)$ must contain one of the three balls A_1, A_2, A_3 in its interior. This implies the lemma: Assume there exists a tetrahedron $xtzw \in \mathcal{D}$ intersecting $\mathcal{B}(y, k_3l)$. The ball $\mathcal{B}(xtzw)$ circumscribed to $xtzw$ obviously also intersects $\mathcal{B}(y, k_3l)$. The prerequisite of the lemma states that each of the A_1, A_2, A_3 contains a point x_j . As we claim, this implies that $\mathcal{B}(xtzw)$ contains one of the A_1, A_2, A_3 and thus the corresponding x_j . This contradicts the assumption that $xtzw$ is a Delaunay tetrahedron.

Consider therefore a ball \mathcal{B} having x on its boundary and intersecting $\mathcal{B}(y, k_3l)$; let o denote its center. Figure 3 illustrates the situation as a projection onto the plane oqy , which is perpendicular to the plane of \mathcal{S} . All the points mentioned in the figure do belong to the plane oqy . (But o is not represented for lack of space.) In particular, the point x , being on the line qy , is part of the figure. Also, let y' be a point of $\mathcal{B} \cap \mathcal{B}(y, k_3l) \cap \text{Plane}(oqy)$. (This intersection is not empty because \mathcal{B} intersects $\mathcal{B}(y, k_3l)$.) The radius k_3l being smaller than the radius k_1l , the line xy' necessarily intersects $\mathcal{B}(q, k_1l)$. (See Figure 3.) The intersection is a segment that we denote cd . By convexity, the ball \mathcal{B} contains xy' and hence also contains cd . This immediately implies that the intersection of the two balls $\mathcal{B} \cap \mathcal{B}(q, k_1l) \stackrel{\text{def}}{=} \mathcal{I}_1$ is not empty. \mathcal{I}_1 is therefore either one of the two balls or the union of two spherical caps. The fact that $k_1 \leq k_3 + k_1 < 1/2$ implies that x does not belong to $\mathcal{B}(q, k_1l)$ and hence that $\mathcal{I}_1 \neq \mathcal{B}$. On the other hand, if $\mathcal{I}_1 = \mathcal{B}(q, k_1l)$ then \mathcal{B} contains all three balls A_1, A_2, A_3 and we are done.

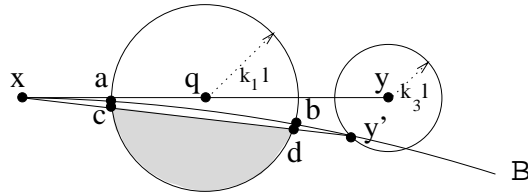


Figure 3: Plane oqy is the plane of the figure (o , the center of ball \mathcal{B} , is not drawn); it is perpendicular to the plane of the spindle \mathcal{S} (that is, the plane of Figure 1). All the points depicted lie in plane oqy ; points x, a, b, y' lie on ball \mathcal{B} ; points x, c, d and y' are collinear. The shaded area represents \mathcal{I}_2 .

We can therefore reduce the analysis to the case of Figure 3: the intersection \mathcal{I}_1 (is convex and) consists of two spherical caps sitting on a disk \mathcal{C}_1 ; this disk is perpendicular to the line segment oq (joining the centers of the two balls) and hence perpendicular to the plane oqy of Figure 3; the latter plane intersects \mathcal{C}_1 in the line segment ab . Consider now another plane, also perpendicular to the plane oqy , but containing the line segment xy' . This plane cuts $\mathcal{B}(q, k_1l)$ in a disk that we denote \mathcal{C}_2 . This disk is also perpendicular to the plane oqy which it intersects in line segment cd . Figure 3 illustrates that, by convexity of the arc $xaby'$, the line segment $xcdy'$ does not intersect the segment ab . This implies that the two disks \mathcal{C}_1 and \mathcal{C}_2 (which are both perpendicular to plane oqy) do not intersect. Now set \mathcal{I}_2 to be the spherical cap of $\mathcal{B}(q, k_1l)$ “below” disk \mathcal{C}_2 , that is, the spherical cap of $\mathcal{B}(q, k_1l)$ cut by \mathcal{C}_2 which does not contain the disk \mathcal{C}_1 . Our discussion allows us to conclude that $\mathcal{I}_2 \subseteq \mathcal{I}_1$.

Recall that, in the previous argument, y' is any point in $\mathcal{B} \cap \mathcal{B}(q, k_3l) \cap \text{Plane}(oqy)$. Therefore, in the case where y belongs to \mathcal{B} we can select y' to be equal to y . In that case \mathcal{I}_2 is an hemisphere of $\mathcal{B}(q, k_1l)$ (which contains trivially one of the balls A_1, A_2, A_3). On the other hand, if $y \notin \mathcal{B}$, then \mathcal{I}_2 is the smaller of the two spherical caps determined by \mathcal{C}_2 .

Summarizing all the previous discussion, we have established so far that, if \mathcal{B} is a ball having x on its boundary and intersecting $\mathcal{B}(y, k_3l)$, then there exists a plane \mathcal{P} containing x , intersecting $\mathcal{B}(y, k_3l)$, and such that the intersection $\mathcal{B} \cap \mathcal{B}(q, k_1l)$ contains the smaller of the two spherical caps cut out of $\mathcal{B}(q, k_1l)$ by \mathcal{P} . Furthermore, it is clear that we can restrict ourselves to the case where \mathcal{P} is tangent to $\mathcal{B}(y, k_3l)$; this corresponds to the case where the spherical cap \mathcal{I}_2 is minimal among all spherical caps cut by planes containing x and intersecting $\mathcal{B}(y, k_3l)$. We can therefore rewrite the preceding summary as follows: Let \mathcal{B} be any ball having x on its boundary and intersecting $\mathcal{B}(y, k_3l)$. Then there exists a plane \mathcal{P}_0 containing x and tangent to $\mathcal{B}(y, k_3l)$ such that the intersection $\mathcal{B} \cap \mathcal{B}(q, k_1l)$ contains the smaller of the two spherical caps cut out of $\mathcal{B}(q, k_1l)$ by \mathcal{P} .

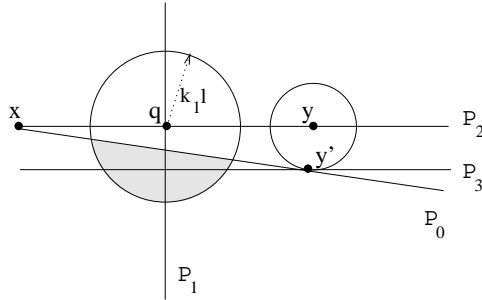


Figure 4: The minimal spherical cap \mathcal{I}_2 cut out of the ball $\mathcal{B}(q, k_1l)$ by the plane \mathcal{P}_0 containing x and tangent to $\mathcal{B}(y, k_3l)$. The plane of the picture is the plane oqy , where o is the center of \mathcal{B} .

Figure 4 illustrates the new situation. \mathcal{P}_0 is such a plane tangent to $\mathcal{B}(y, k_3l)$, and \mathcal{I}_2 is the smaller of the two spherical caps that \mathcal{P}_0 cuts off $\mathcal{B}(q, k_1l)$. Now let \mathcal{P}_2 be the plane perpendicular to the plane oqy (the plane of the figure) which contains the line xqy , and let \mathcal{P}_3 be the plane parallel to \mathcal{P}_2 and tangent to $\mathcal{B}(y, k_3l)$; obviously, the distance from q to \mathcal{P}_3 is equal to k_3l . Finally, let \mathcal{P}_1 denote the plane that contains

q and is perpendicular to \mathcal{P}_2 . Note that \mathcal{P}_1 is also the plane of Figure 1 defined by the centers of the three balls A_1, A_2 and A_3 of the spindle \mathcal{S} . Moreover, the planes \mathcal{P}_2 and \mathcal{P}_3 intersect \mathcal{P}_1 as indicated on Figure 1.

To come to a conclusion, let \mathcal{I}_3 denote the smaller of the two spherical caps cut by \mathcal{P}_3 into $\mathcal{B}(q, k_1 l)$. Our earlier choices for k_1, k_2, k_3 imply that \mathcal{I}_3 contains at least one of the three balls A_1, A_2, A_3 of the spindle \mathcal{S} . Since $\mathcal{I}_3 \subseteq \mathcal{I}_2 \subseteq \mathcal{I}_1$, we established that the ball \mathcal{B} contains at least one of the three balls of the spindle, and Lemma 3 is proven. ■

4.2 A Set of Probabilistic Lemmas

In order to prove the theorems, we need a set of probabilistic lemmas. The proof of Lemma 5 hinges critically on Lemma 3.

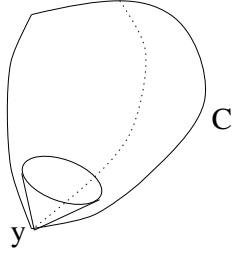


Figure 5: For every point y of a convex solid C and every small r , C contains a spherical cone of radius r and of solid angle θ issued from y .

Lemma 4. Let X_1 be a random variable drawn from an (α, β) -measure over a bounded convex set C . Then there exist constants $r_0 > 0$ and $\gamma > 0$ such that for every $r \leq r_0$,

$$\inf_{y \in C} P[d(X_1, y) \leq r] \geq \gamma r^3.$$

Proof: By convexity of C there exists a distance r_0 and a value $\theta \leq 4\pi$ such that the following holds. Let y be any point in C . Then there exists a spherical cone \mathcal{C} with summit y , radius r and solid angle θ such that \mathcal{C} is included in C . The volume $\text{Vol}(\mathcal{C})$ of this cone is $\frac{\theta}{4\pi} r^3$. Therefore, if X_1 is a random variable drawn from an (α, β) -measure over C , we have:

$$P[d(X_1, y) \leq r] \geq P[X_1 \in \mathcal{C}] \geq \alpha \text{Vol}(\mathcal{C}) = \alpha \frac{\theta}{4\pi} r^3.$$

■

In the following, we denote the set of all Delaunay tetrahedra incident to a point X_i by \mathcal{T}_{X_i} . Lemma 5 estimates N_1 , the number of X_i 's such that \mathcal{T}_{X_i} intersects $\mathcal{B}(y, r)$; in other words, the number of X_i 's with the property that at least one of its incident Delaunay tetrahedra intersects $\mathcal{B}(y, r)$. With this, Lemma 6 will estimate the number of Delaunay tetrahedra intersecting $\mathcal{B}(y, r)$.

Lemma 5. Consider n points X_1, \dots, X_n drawn independently from an (α, β) -measure over a bounded convex set C with small curvature. Then there exist positive

constants a, b, c and d , depending upon α, β and C only, such that the following holds. Let y be any point in C at distance $a(\log n/n)^{1/3}$ from the boundary ∂C and $r \leq 1/2 d(y, \partial C)$ be a positive quantity. Let N_1 denote the number of X_i 's with the property that one of the tetrahedra in \mathcal{T}_{X_i} intersects $\mathcal{B}(y, r)$. Then

$$E[N_1] \leq b + c r^2 n^{2/3} + d n r^3.$$

Proof: By linearity of the expectation, $E[N_1] = np$, where p is the probability that one of the Delaunay tetrahedra incident to X_1 intersects $\mathcal{B}(y, r)$. Let $L = d(X_1, y)$; L is itself a random variable. In the following we condition on the value of X_1 . The points X_1 and y are then fixed, and we let \mathcal{S}_{X_1} be any L -spindle whose axle is $X_1 y$. (Hence the angle θ of Figure 1 is arbitrary but fixed.) As before, A_1, A_2, A_3 denote the 3 balls of the wheel. Remark that the event $\{\mathcal{S}_{X_1} \subseteq C\} = \{\text{the 3 balls of the spindle } \mathcal{S}_{X_1} \text{ are included in } C\}$.

In the following, let event $B_1 = \{\mathcal{S}_{X_1} \subseteq C, k_3 L \geq r\}$. Note that the relation $k_3 L \geq r$ implies that X_1 does not belong to $\mathcal{B}(y, r)$. The first inequality is therefore a direct consequence of Lemma 3. In the third inequality, $\mathbf{1}_{k_2 L \leq r_0}$ denotes the random variable equal to 1 when $k_2 L \leq r_0$ and 0 else.

$$\begin{aligned} & P[\mathcal{T}_{X_1} \text{ intersects } \mathcal{B}(y, r) \mid X_1, B_1] \\ & \leq P[\text{one of the 3 balls } A_1, A_2, A_3 \text{ contains no point } X_2, \dots, X_n \mid X_1, B_1] \\ & \leq \sum_{j=1}^3 P[A_j \text{ contains none of the points } X_2, \dots, X_n \mid X_1, B_1] \\ & = \sum_{j=1}^3 \prod_{i=2}^n P[X_i \notin A_j \mid X_1, B_1] \\ & \leq 3E\left[(1 - \gamma(k_2 L)^3)^{n-1} \mathbf{1}_{k_2 L \leq r_0} \mid X_1, B_1\right] \\ & \quad + 3E\left[(1 - \gamma r_0^3)^{n-1} \mathbf{1}_{k_2 L > r_0} \mid X_1, B_1\right]. \end{aligned} \tag{1}$$

Equation (1) is a consequence of the (conditional) independence of the events $X_i \notin A_j$. This independence comes both from the fact that the random variables X_i are independent and the fact that the 3 balls A_1, A_2, A_3 are defined *independently* of the points X_2, \dots, X_n . The necessity of this last independence is not always recognized and leads to frequent mistakes in the literature.

We now justify the last inequality. Note first that, by Lemma 4, $\gamma(k_2 L)^3 \leq 1$ when $k_2 L \leq r_0$. The expression $(1 - \gamma(k_2 L)^3)^{n-1}$ is therefore well-defined. The conditioning on $\mathcal{S}_{X_1} \subseteq C$ ensures that each ball A_i is fully contained in C . This implies in particular that the center y_i of each ball A_i is in C . We can then apply Lemma 4 (which determines the values γ and r_0) using the fact that the random variables X_i are all drawn independently according to an (α, β) -measure.

We let $\overline{B_1} = \{k_3 L < r\} \cup \{\mathcal{S}_{X_1} \not\subseteq C\}$ denotes the complement of B_1 . Integrating the previous majoration of $P[\mathcal{T}_{X_1} \text{ intersects } \mathcal{B}(y, r) \mid X_1, B_1]$ with respect to X_1 therefore gives:

$$\begin{aligned} p &= P[\mathcal{T}_{X_1} \text{ intersects } \mathcal{B}(y, r), \overline{B_1}] + P[\mathcal{T}_{X_1} \text{ intersects } \mathcal{B}(y, r), B_1] \\ &\leq P[k_3 L < r] + P[\mathcal{T}_{X_1} \text{ intersects } \mathcal{B}(y, r), \mathcal{S}_{X_1} \not\subseteq C] \end{aligned}$$

$$\begin{aligned}
& + 3 E \left[\left(1 - \gamma(k_2 L)^3 \right)^{n-1} \mathbf{1}_{k_2 L \leq r_0} \mathbf{1}_{k_3 L \geq r} \right] \\
& + 3 \left(1 - \gamma r_0^3 \right)^{n-1} P[k_2 L > r_0; k_3 L \geq r] \\
& \stackrel{\text{def}}{=} I + II + III + IV.
\end{aligned}$$

The fact that X_1 is drawn from an (α, β) -measure implies that $I = P[k_3 L < r] \leq (4/3) \beta \pi (r/k_3)^3$. Also $IV \leq 3e^{-(n-1)\gamma r_0^3}$, which is exponentially small with n sufficiently large. We now turn to III . Note first that $(1 - \gamma(k_2 L)^3)^{n-1} \mathbf{1}_{k_2 L \leq r_0} \leq e^{-(n-1)\gamma(k_2 L)^3}$ and therefore, $III \leq 3E \left[e^{-(n-1)\gamma(k_2 L)^3} \mathbf{1}_{k_3 L \geq r} \right]$. To estimate this expression we use spherical coordinates and obtain

$$III \leq 16\pi^2 \beta \left(\frac{2}{3(n-1)k_2^3 \gamma} + \frac{2r^2}{k_2 k_3^2 (\gamma(n-1))^{1/3}} \frac{1}{3} \Gamma\left(\frac{1}{3}\right) \right).$$

We now turn to expression $II = P[\mathcal{T}_{X_1} \text{ intersects } \mathcal{B}(y, r), S_{X_1} \not\subseteq C]$; its majoration will involve showing that only “local” vertices X_1 have a Delaunay tetrahedron extending to y .² We will use this general fact in the vicinity of the boundary ∂C of C . Part of our assumptions is that C has low curvature. Therefore, at the very small distances that we consider, ∂C appears flat. We will take advantage of this fact and model locally the boundary ∂C to be a plane \mathcal{P} : C appears locally like a half-plane U .³

To simplify we set $K = 2k_1$ and recall that $K < 1 - 2k_3 < 1$. Recall also that we defined S_{X_1} to be any arbitrary, externally fixed, spindle whose axle is yX_1 : S_{X_1} is not uniquely determined by X_1 but also by the angle θ of Figure 1. For every X_1 define S'_{X_1} to be the 3-dimensional “tire” span by S_{X_1} when rotating around its axle. S'_{X_1} is *uniquely* determined by X_1 (along with y, K) and contains S_{X_1} so that, clearly, $\{S_{X_1} \not\subseteq U\} \subseteq \{S'_{X_1} \not\subseteq U\}$. To further simplify, introduce S''_{X_1} to be the following (simpler) object. S''_{X_1} is composed of (i) the axle $X_1 y$, and (ii) a circle of diameter K , the *wheel*, perpendicular to $X_1 y$ whose center is the mid-point q of $X_1 y$. We sometimes write $S''_{X_1}(K)$ to emphasize the value of K . Furthermore, let \mathcal{P} be a plane such that $y \notin \mathcal{P}$, let U denote the \mathcal{P} -half-space to which y belongs, and let h denote the distance $d(y, \mathcal{P})$ from y to \mathcal{P} . We only consider points X in U and say that the spindle S''_{X_1} *crosses* \mathcal{P} if its wheel crosses \mathcal{P} . As before, we set $L = d(X_1, y)$.

Claim: (a) $S''_{X_1}(K)$ crosses \mathcal{P} only if $L \geq L_0(h) \stackrel{\text{def}}{=} \sqrt{1 + 1/K^2} h$.
(b) If $K \leq 0.2$ then $\{S'_{X_1}(K) \not\subseteq U\} \subseteq \{S''_{X_1}(1.02K) \not\subseteq U\}$.

We just show (a). Assertion (b) implies that, for k_1 small enough, a minute adjustment in k_1 allows to consider $\{S''_{X_1} \not\subseteq U\}$ in place of $\{S'_{X_1} \not\subseteq U\}$.

Consider a given value of L and assume that an L -spindle S''_{x_1} crosses \mathcal{P} . Let m be a point on the wheel “below” \mathcal{P} ; see Figure 6. Note that $k_1 < 1/2$ implies that $d(y, \mathcal{P}) \leq d(y, m) = (d(y, q)^2 + d(q, m)^2)^{1/2} < L/\sqrt{2}$. This implies that there exists a point $x_0 \in \mathcal{P}$ and such that $d(y, x_0) = L$. (This point can be obtained by rotating the spindle around y .) Consider S''_{x_0} , the spindle associated to x_0 . The fact that S''_{x_1} crosses \mathcal{P} implies that S''_{x_0} similarly crosses \mathcal{P} . (We provide a pictorial justification of

²Here, “local” means being within distance $O((\log n/n)^{1/3})$ of y : a quantity asymptotically small.

³This is valid only locally. Recall: C is bounded.

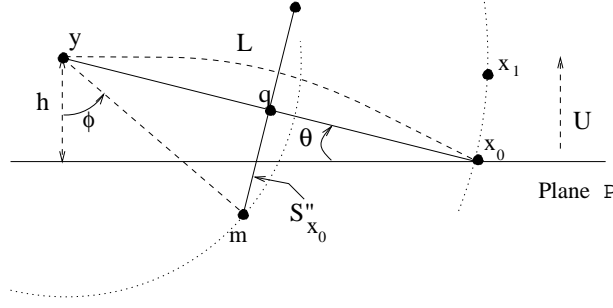


Figure 6: A spindle \mathcal{S}''_{x_0} crossing \mathcal{P} .

this fact. Consider the rotation of the spindle around y in the plane of Figure 6: we let ϕ denote the angle between ym and the vertical line; ϕ_1 is the angle corresponding to the initial point x_1 ; ϕ_0 is the angle corresponding to x_0 ; we also let m_1 and m_0 denote the positions of m associated respectively to x_1 and x_0 . Obviously m goes “deeper” below \mathcal{P} as $\phi \in [0, \pi]$ decreases to 0. As $d(y, q) > d(q, m)$ the angle ϕ_0 belongs to $[0, \pi/2]$. Therefore m_0 is deeper below \mathcal{P} than m_1 .)

To prove impossibility results we can therefore restrict ourselves to $x \in \mathcal{P}$. Consider such an x . The fact that \mathcal{S}''_x crosses \mathcal{P} implies that $d(q, m) = k_1 L \geq L/2 \tan \theta$ where $\theta = \arcsin(h/L)$. This immediately implies $L^2 \geq \frac{1+K^2}{K^2} h^2$. We are now ready for the majorations below.

$$\begin{aligned}
& P[\mathcal{S}''_{X_1} \not\subseteq U, \mathcal{T}_{X_1} \text{ intersects } \mathcal{B}(y, r)] \\
&= P[\mathcal{S}''_{X_1} \not\subseteq U, \exists X_{i_2}, X_{i_3}, X_{i_4} \text{ such that } X_1 X_{i_2} X_{i_3} X_{i_4} \in \mathcal{D} \\
&\quad \text{and such that } X_1 X_{i_2} X_{i_3} X_{i_4} \cap \mathcal{B}(y, r) \neq \emptyset] \\
&\leq \max_{y' \in \mathcal{B}(y, r)} P[\exists X_{i_2}, X_{i_3}, X_{i_4} ; d(X_1, y) \geq L_0(h), X_1 X_{i_2} X_{i_3} X_{i_4} \in \mathcal{D}, \\
&\quad y' \in \mathcal{B}(X_1 X_{i_2} X_{i_3} X_{i_4})] \\
&\leq \max_{y' \in \mathcal{B}(y, r)} \binom{n-1}{3} P[d(X_1, y) \geq L_0(h), X_1 X_2 X_3 X_4 \in \mathcal{D}, y' \in \mathcal{B}(X_1 X_2 X_3 X_4)] \\
&\leq \max_{y' \in \mathcal{B}(y, r)} n^3/6 \int_{T_{y'}} dP_{X_1, X_2, X_3, X_4}(x, t, z, u) (P[X_5 \notin \mathcal{B}(xtzu)]^{n-4}),
\end{aligned}$$

where $T_{y'} \stackrel{\text{def}}{=} \{(x, t, z, u) \in U^4; d(x, y) \geq L_0(h), \text{ and } y' \in \mathcal{B}(xtzu)\}$, and P_{X_1, X_2, X_3, X_4} denotes the probability law of the random variable (X_1, X_2, X_3, X_4) .

We now compute an upper-bound for the expression $P[X_5 \notin \mathcal{B}(xtzu)]$ when the points x, t, z, u of U are such that $d(x, y) \geq L_0(h)$, and such that $y' \in \mathcal{B}(xtzu)$. We define $K' \stackrel{\text{def}}{=} \sqrt{1 + 1/K^2}$. Hence $L_0(h) = K'h$. We also set $K'' \stackrel{\text{def}}{=} 2K' - 1$. We need the following claim.

Claim: If $d(x, y) \geq L_0(h)$ and $y' \in \mathcal{B}(xtzu) \cap \mathcal{B}(y, r)$ then $\text{Vol}(\mathcal{B}(xtzu) \cap U) \geq \frac{\pi}{24} (\frac{K''^2}{1-1/K''^2})^{1/2} h^3 \stackrel{\text{def}}{=} c'h^3$.

The situation is presented in Figure 7. The ball $\mathcal{B}(xtzu)$ cuts minimally U only (i) when x is on the boundary \mathcal{P} , (ii) when y' is on the boundary of $\mathcal{B}(xtzu)$, and (iii)

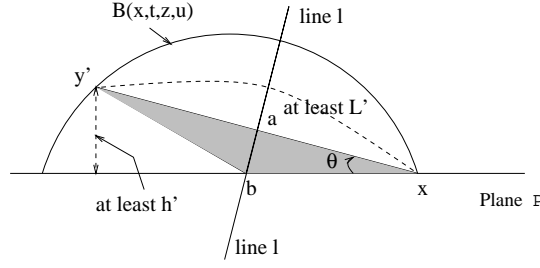


Figure 7: The cone \mathcal{C} contained in every $\mathcal{B}(xtzu)$, containing y' and such that $d(x, y') \geq L_0(h)$.

when the center o of $\mathcal{B}(xtzu)$ is such that the plane xoy' is perpendicular to \mathcal{P} . We therefore consider the situation within plane xoy' , as in Figure 7.

The center o is located on the line l perpendicular to $y'x$ and going through the mid-point a of $y'x$. Let b be the intersection of l with \mathcal{P} . Consider the cone \mathcal{C} issued from b and whose base is the circle with diameter $y'x$. We claim that \mathcal{C} is included in the ball $\mathcal{B}(xtzu)$ cutting minimally U . (i) This ball $\mathcal{B}(xtzu)$ must have its center o below a : if not, at least half of $\mathcal{B}(xtzu)$ is in U . (ii) If o is under b , then b is in the convex hull of the 3 points y', x, o . All 3 points y', x, o are in $\mathcal{B}(xtzu)$ and hence so is b . $\mathcal{B}(xtzu)$ then clearly contains \mathcal{C} . (iii) If c belongs to the segment ab then $d(o, b) \leq k_1 L < L/2 \leq d(o, y')$. Thus, if $o \in [a, b]$ then $b \in \mathcal{B}(xtzu)$ and $\mathcal{B}(xtzu)$ contains \mathcal{C} as before.

This shows that $v \stackrel{\text{def}}{=} \text{Vol}(\mathcal{B}(xtzu) \cap U) \geq \text{Vol}(\mathcal{C})$. Using the fact that $d(y, y') \leq r \leq h/2$, we find that v is at least equal to $\frac{\pi}{3}(L'/2)^3 \tan(\theta'/2)$, where $h' \stackrel{\text{def}}{=} h - h/2 = h/2$, $L' \stackrel{\text{def}}{=} L_0(h) - h/2 = K''h'$ and where $\theta' = \arcsin(h'/L')$. We obtain $v \geq c'h^3$, where $c' = \frac{\pi}{24}(\frac{K''^2}{1-K''^2})^{1/2}$, and have thus proven:

$$P[S''_{X_1} \not\subseteq \mathcal{C}, \mathcal{T}_{X_1} \text{ intersects } \mathcal{B}(y, r)] \leq \frac{n^3}{6}(1 - c'\alpha h^3)^{n-4} \leq \frac{n^3}{6}e^{-(n-4)c'\alpha h^3}.$$

Selecting $h(n) = \left(\frac{4}{c'\alpha} \frac{\log n}{n}\right)^{1/3}$ gives $II = O(1/n)$, as needed. Then, multiplying by n the bounds found for I, II, III and IV and summing establishes the result, and concludes the proof of Lemma 5. \blacksquare

Lemma 6. Under the hypothesis of Lemma 5

$$E[N_2] \leq e(b + cr^2n^{2/3} + dnr^3) \log n / \log \log n,$$

where N_2 is the number of Delaunay tetrahedra that intersect $\mathcal{B}(y, r)$ and where e is a constant depending solely on C, α and β .

To prove this, we need the following result, very similar to the result derived by [BEY91] in their Theorem 7.

Lemma 7. There exists a constant c'' such that $P[d^o(X_1) > c'' \log n / \log \log n] \leq 1/n^4$. Therefore

$$P[\exists i, d^o(X_i) > c'' \log n / \log \log n] \leq 1/n^3.$$

Proof: The proof of this result follows very closely the proof for the Poisson model given in Lemmas 8 and 9 of [BEY91]. The only technical difference is that they bound the first probability by $1/n^2$ instead of $1/n^4$. We show here that $1/n^4$ is similarly valid. A careful reading of their proof shows that we only need to establish that, with probability at least $1 - 1/n^4$, the maximum Delaunay edge length is $O((\log n/n)^{1/3})$.⁴

We compute:

$$\begin{aligned} P[d(X_1, X_2) \geq h; X_1 X_2 \in \mathcal{D}] \\ &\leq P[d(X_1, X_2) \geq h; \exists i_3, i_4 \text{ such that } X_1 X_2 X_{i_3} X_{i_4} \in \mathcal{D}] \\ &\leq n^2(1 - \gamma(h/2)^3)^{n-d-1} \leq O(n^2 e^{-\gamma(h/2)^3 n}). \end{aligned}$$

Therefore, $P[\exists X_{i_1} X_{i_2} \text{ such that } d(X_{i_1}, X_{i_2}) \geq h \text{ and } X_{i_1} X_{i_2} \in \mathcal{D}] \leq O(n^4 e^{-\gamma(h/2)^3 n})$.

This is $o(1/n^4)$ if $h > \left(\frac{32 \log n}{\gamma n}\right)^{1/3}$. \blacksquare

Proof of Lemma 6: By Euler's formula, there is a constant K such that the tetrahedron-degree is equal to K times the edge-degree. For convenience, assume that B_2 denotes the event $\{\forall i, d^o(X_i) \leq c'' \log n / \log \log n\}$, which is identical to the event $\{\forall i, \text{tetrahedron-}d^o(X_i) \leq K c'' \log n / \log \log n\}$. Then, from Lemma 7 it follows that $P[\overline{B_2}] \leq 1/n^3$ and we have: $E[N_2] = E[N_2; B_2] + E[N_2; \overline{B_2}]$. The two terms $E[N_2; B_2]$ and $E[N_2; \overline{B_2}]$ need to be bounded separately. We begin with $E[N_2; B_2]$:

$$\begin{aligned} E[N_2; B_2] &= P[B_2] E[N_2 | B_2] \\ &\leq K c'' P[B_2] E[N_1 | B_2] \log n / \log \log n \\ &= K c'' E[N_1; B_2] \log n / \log \log n \\ &\leq K c'' E[N_1] \log n / \log \log n \\ &\leq K c'' (b + c r^2 n^{2/3} + d n r^3) \log n / \log \log n. \end{aligned}$$

On the other hand, $E[N_2; \overline{B_2}] \leq O(n^2) P[\overline{B_2}] \leq O(n^2) 1/n^3 = o(1)$. \blacksquare

Corollary 8. Consider n points X_1, \dots, X_n drawn independently from an (α, β) -measure over a bounded convex set C with small curvature. Then there exist positive constants a, b, c, d and e , depending upon α, β and C only, such that the following holds. Let y be any point in C at distance $a(\log n/n)^{1/3}$ from the boundary ∂C . Let $r, r \leq \frac{a}{2n^{1/3}}$ be a positive quantity. Let N_2 denote the number of Delaunay tetrahedra that intersect $\mathcal{B}(y, r)$. Then

$$E[N_2] \leq e(b + c(a/2)^2 + d(a/2)^3) \log n / \log \log n.$$

4.3 Summing It All Up

We are ready to prove the theorems.

⁴Theorem 1 of [BEY91] establishes that the maximum edge length is at most $O((\log n)^{1/3})$ with high probability. Our additional factor $1/n^{1/3}$ comes from the fact that they consider a cube of variable side length $1/n^{1/3}$.

Proof of Theorem 2: We set $c_1 = a$, where a is the constant of Lemma 5 and Corollary 8. The segment L may be covered by $\left\lceil \frac{|L|}{a} n^{1/3} \right\rceil$ circles of radius $\frac{a}{2n^{1/3}}$ each and centered on points y_i of L . The number N of intersections between L and the Delaunay triangulation is bounded by the sum of the number of intersections with these circles. By Corollary 8, the expected number of intersections with each of these circles is bounded by $K \log n / \log \log n$ for some constant K depending upon α, β and C only. Hence,

$$\begin{aligned} E[N] &\leq K \left\lceil \frac{|L|}{a} n^{1/3} \right\rceil \log n / \log \log n \\ &\leq K \left(1 + \frac{|L|}{a} n^{1/3} \right) \log n / \log \log n \end{aligned}$$

which proves Theorem 2. \blacksquare

Proof of Theorem 1: We have in mind to apply Theorem 2 to the segment $L = qY$. We are faced with two difficulties. First, both Y and L are defined in terms of Y_1, \dots, Y_m and are therefore not independent of X_1, \dots, X_n . Second, Y can possibly be within distance $c_4(\log n/n)^{1/3}$ from the boundary ∂C . We will solve the first difficulty by considering a slightly different Delaunay triangulation with respect to which L is independent. We will solve the second difficulty by showing that Y is with high probability at distance of at least $c_4(\log n/n)^{1/3}$ from ∂C .

Let us first recall that q and Y are defined in very different ways. The condition that they be “far enough” from the boundary must therefore be handled differently. The query point q is not in the control of the algorithm. It is instead decided externally and the algorithm is claimed to perform well for all admissible choices of q . Thus, the assumption “ q is at distance of at least $n^{1/18}$ from ∂C ” is merely a restriction on the set of query points against which the algorithm has to measure. On the other hand, the point Y is chosen randomly, as described in the algorithm on page 2. The fact that “ Y is at distance of at least $c_4(\log n/n)^{1/3}$ from ∂C ” cannot therefore be imposed externally.

Let us relabel $\{X_1, \dots, X_n\} - \{Y_1, \dots, Y_m\}$ into $\{X'_1, \dots, X'_{n-m}\}$. As usual, let \mathcal{D} denote the Delaunay triangulation associated to the n points X_1, \dots, X_n , and let \mathcal{D}' denote the Delaunay triangulation associated to the $n-m$ points X'_1, \dots, X'_{n-m} . The random variables X'_1, \dots, X'_{n-m} are independent from the random variables Y_1, \dots, Y_m . This implies that, for every query point q , (X'_1, \dots, X'_{n-m}) is independent from Y , which allows us to make the following two conclusions. First, $L \stackrel{\text{def}}{=} (Y, q)$, the line segment connecting Y and q , is independent of the $n-m$ data points defining \mathcal{D}' . Second, the probabilistic behavior of X'_1, \dots, X'_{n-m} is unaffected when conditioning on the event $B_3 \stackrel{\text{def}}{=} \{d(Y, \partial C) \geq c_4(\log n/n)^{1/3}\}$. In formal terms, the probabilistic law $\mathcal{L}(X'_1, \dots, X'_{n-m})$ is equal to the conditional law $\mathcal{L}((X'_1, \dots, X'_{n-m}) | B_3)$. In particular, the random variables X'_1, \dots, X'_{n-m} are independent identically distributed (α, β) -random variables under the conditional probability distribution $P[\cdot | B_3]$.

Let N denote the number of tetrahedra in \mathcal{D}' crossed by L . We have $E[N] = E[N; B_3] + E[N; \overline{B_3}]$ where $\overline{B_3}$ denotes the complement of B_3 . We provide upper bounds for the two terms $E[N; B_3]$ and $E[N; \overline{B_3}]$.

We begin with $E[N; \overline{B_3}]$. It is well known that $N = O(n^2)$. Hence $E[N; \overline{B_3}] \leq$

$O(n^2)P[\overline{B_3}]$. We fix c to be a constant such that, for every n , $c_4(\log n/n)^{1/3} \leq \frac{c}{2}(1/n)^{1/18}$. Recall that $P[\overline{B_3}] = P[d(Y, \partial C) < c_4(\log n/n)^{1/3}]$ and that, by assumption, $d(q, \partial C) \geq c(1/n)^{1/18}$. By triangular inequality, this implies that:

$$\begin{aligned}
P[\overline{B_3}] &\leq P\left[d(Y, q) \geq \frac{c}{2} \frac{1}{n^{1/18}}\right] \\
&= \left(P\left[d(Y_1, q) \geq \frac{c}{2} \frac{1}{n^{1/18}}\right]\right)^m \\
&= \left(1 - P\left[d(Y_1, q) \leq \frac{c}{2} \frac{1}{n^{1/18}}\right]\right)^m \\
&\leq e^{-m P\left[d(Y_1, q) \leq \frac{c}{2} \frac{1}{n^{1/18}}\right]} \\
&\leq e^{-n^{1/5} \alpha^{\frac{4\pi}{3}} \left(\frac{c}{2}\right)^3 \frac{1}{n^{1/6}}} \\
&= o(1/n^2).
\end{aligned}$$

This shows that $E[N; \overline{B_3}] = o(1)$. We now turn to $E[N; B_3] = E[N | B_3] P[B_3]$. Theorem 2 (see the remark after Theorem 2), along with the fact that X'_1, \dots, X'_{n-m} are (α, β) -random variables, independently identically distributed under the measure $P[\cdot | B_3]$, implies that:

$$E[N | B_3] \leq c_4 + c_5 E[d(Y, q) | B_3] (n-m)^{1/3} \log n / \log \log n.$$

Hence,

$$\begin{aligned}
E[N; B_3] &\leq c_4 P[B_3] + c_5 E[d(Y, q); B_3] (n-m)^{1/3} \log n / \log \log n \\
&\leq c_4 + c_5 E[d(Y, q)] n^{1/3} \log n / \log \log n.
\end{aligned}$$

The estimation of $E[d(Y, q)]$ is done as in [DMZ95]. The beginning of the argument is similar to the estimation of $P[\overline{B_3}]$ above. Lemma 4 is then used. We let $\text{diam}(C)$ denote the diameter of C . Note that Y and q are in C so that $P[d(Y, q) > t] = 0$ if $t > \text{diam}(C)$.

$$\begin{aligned}
E[d(Y, q)] &= \int_0^\infty P[d(Y, q) > t] dt \\
&\leq \int_0^\infty e^{-m P[d(Y_1, q) \leq t]} dt \\
&\leq \int_0^{r_0} e^{-m \alpha \gamma t^3} dt + \int_{r_0}^{\text{diam}(C)} e^{-m \alpha \gamma r_0^3} dt \\
&\leq \int_0^\infty e^{-m \alpha \gamma t^3} dt + \text{diam}(C) e^{-m \alpha \gamma r_0^3} \\
&= O\left(\frac{1}{m^{1/3}}\right)
\end{aligned}$$

We have therefore shown that $E[N] = O((n/m)^{1/3} \log n / \log \log n)$.

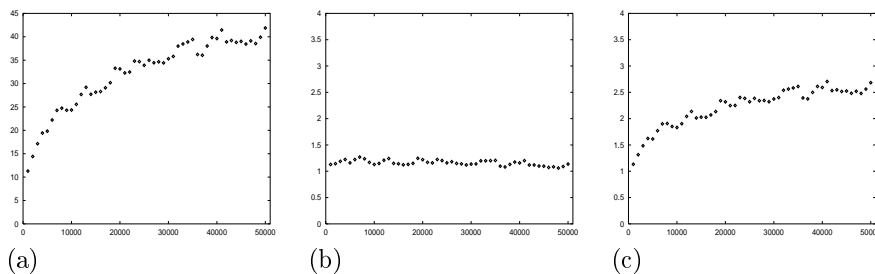


Figure 8: (a) The sample mean M_n of the number of triangles visited, for a sample of 999 random query points q for each data set of size n . (b) The ratio $M_n/n^{1/3}$. (c) The ratio $M_n/\log_2 n$.

N_{total} , the total number of tetrahedra in \mathcal{D} crossed by L is not more than that for \mathcal{D}' , that is, above N , plus the sum S of the tetrahedra degrees of (that is, the number of tetrahedra adjacent to) Y_1, \dots, Y_m in the Delaunay triangulation \mathcal{D} . To see this, note that L either crosses a tetrahedron without one of the Y_i 's as a vertex (in which case the tetrahedron is both in \mathcal{D} and \mathcal{D}') or one for which Y_i is a vertex (in which case the tetrahedron is in \mathcal{D} but not in \mathcal{D}'). The total number of the latter kind of tetrahedra does not exceed S . The expected value of S is, by linearity of expectation, $3m$ times the expected (vertex) degree $\delta(n)$ of Y_1 , where the constant 3 results from Euler's formula. Combining all this we have:

$$E[N_{\text{total}}] = O\left((n/m)^{1/3} \log n / \log \log n + m\delta(n)\right).$$

The time complexity T of the jump-and-march algorithm on page 2 is proportional to $m + N_{\text{total}}$; the sample size m comes into play because of steps (1) and (2), N_{total} is due to step (3). $E[T]$ can thus be optimized to $O(\delta(n)^{1/4} n^{1/4} (\log n / \log \log n)^{3/4})$ with the choice of $m = \Theta(n^{1/4} / \delta(n)^{3/4} (\log n / \log \log n)^{3/4})$. ■

5 Empirical Results in 2D

This section presents some empirical results on the planar jump-and-march, or better, a variation of it. For further convenience, we sample $n^{1/3}$ edges of the Delaunay triangulation \mathcal{D} , rather than points. Then, we choose the edge whose midpoint has minimum distance to the query point q . We find the triangle containing q by traversing the triangles intersected by $L = (y, q)$, where y is the midpoint of the initially chosen edge.

We tested this procedure for random point sets of size $n = 1000, 2000, \dots, 50000$; the coordinates were chosen randomly out of the unit square. In Figures 8 (a) and 8 (b), M_n denotes the sample mean of the number of triangles visited, over a sample of 999 queries, and for one random point set of size n , for each n ; the coordinates of q (and the point set) are chosen by random out of the unit square. Thus, M_n corresponds to the $E[N_{\text{total}}]$ in the analysis. Since $\Theta(\log n)$ is the best known theoretical bound for planar point locations; see, for example, [PS85], Figure 8 (c) plots the ratio $M_n / \log_2 n$ to give a measure for the efficiency of the method. Note that the best known planar

point location algorithm [DL76] is obtained by two binary searches, one horizontally and one vertically, thus has at least a constant of 2 in front of the $\log_2 n$.

It should be noted that it might be difficult to compare our algorithm with that of the bucketing algorithm of [AEI⁺85]. Although the latter algorithm takes an average constant query time, the constant depends on the size of the buckets, hence depends on the amount of preprocessing performed in the buckets. We believe that when n is significantly big, for example, n is greater than a million, the bucketing method might be the best solution for planar point locations. In any case, remember that bucketing requires preprocessing and additional data structures.

6 Empirical Results in 3D

How does the jump-and-march perform in 3D? Again, we implement a slight variation of the procedure analyzed in this text; the empirical studies listed here, but also in the context of incremental Delaunay triangulators [M93], justify this. In order to discuss the procedure, it helps to have the concept of an *oriented triangle* τ . In essence, it is given by the (ordered) sequence of its vertices. The vertices then define (the underlying plane with) a normal vector pointing, by definition, to the triangle's *positive side*, denoted by τ^+ . Linear algebra dictates that a triangle has two distinguishable orientations.

With this, our variant, the *jump-and-walk*, works as follows. First, we sample triangles rather than vertices. The sample size is set to $m = O(n^{1/4})$, for Delaunay triangulations of n points. The “distance” of a triangle to the query point q is calculated as the minimum distance of its three vertices to q . The triangle τ_0 which scores with the shortest distance is selected. We adjust its orientation such that q is on its positive side, that is, $q \in \tau_0^+$. The selection of τ_0 constitutes the “jump” part of the algorithm.

Second, we perform the following loop, which implements to original “walking” strategy mentioned in the introduction. The loop has the invariant that $q \in \tau^+$.

- (1) If τ is a convex hull triangle and τ^+ is outside \mathcal{D} , then q lies outside \mathcal{D} .
Exit loop.
- (2) Otherwise, there is a tetrahedron in \mathcal{D} incident to τ (and in τ^+).
If this tetrahedron contains q , exit loop.
- (3) Otherwise, select a triangle σ of the tetrahedron, such that, $q \in \sigma^+$.
- (4) Set $\tau = \sigma$, and continue loop at (1).

Each iteration of this loop, corresponds to a tetrahedron visited. Again, note that this procedure is only guaranteed to terminate for Delaunay triangulations since they are proven to be “acyclic for any fixed viewpoint” [E90]. For arbitrary triangulations, this is not necessarily the case; however, if the selection of σ in (3) is done by random (out of the up to 3 possibilities), then the infinite loop is broken with probability arbitrary close to 1.

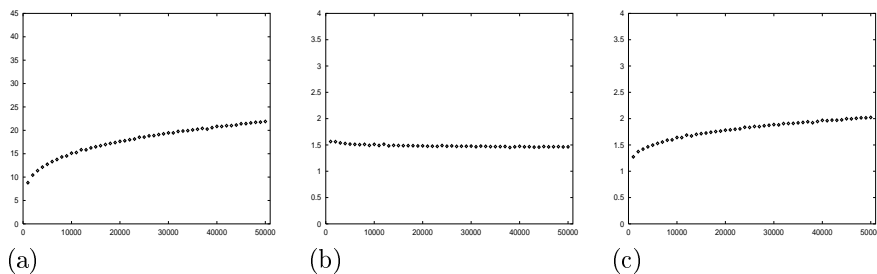


Figure 9: (a) The sample mean M_n of the number of tetrahedra visited, for a sample of 9999 random query points q for a random point set of size n . The plotted value M_n reflects the worst mean out of 5 random point sets for each n . (b) The ratio $M_n/n^{1/4}$. (c) The ratio $M_n/\log_2 n$.

It is clear that, in terms of the number of tetrahedra visited, jump-and-walk can only be worse than jump-and-march. However, since orientation tests in 3D are computationally less expensive than intersection tests (in particular, if implemented in a provable robust fashion), jump-and-walk will still be superior in terms of CPU time.

For our experiments, we generated 5 random 3D point sets for each $n = 1000, 2000, \dots, 50000$. Each data set was then queried with 9999 random points, using the jump-and-walk algorithm. We chose $m = 7n^{1/4}$ for the random sample. This was empirically determined to be the best choice with respect to actual CPU time. It is obvious that the larger m , the smaller the number of tetrahedra visited in the walk. For $m = 2n^{1/4}$, these numbers roughly balance. However, it makes sense to increase m , because the sampling of a triangle in the jump phase is computationally less expensive than the visit of a tetrahedron while walking towards the query point. This is so in particular because care has to be taken to implement the walk robustly, for example, using symbolic perturbation and exact arithmetic.

For each of the 9999 query points we count the number of tetrahedra visited and take the mean. This gives us 5 sample mean values for each n ; Figure 9 (a) plots M_n , that is, the largest of the 5. The corresponding 90% confidence intervals were consistently smaller than $\pm 1\%$. Here, we say that the sample mean $\hat{\mu}$ has a 90% *confidence interval* of $\pm H\%$ if the interval $[\hat{\mu}(1 - \frac{H}{100}), \hat{\mu}(1 + \frac{H}{100})]$ contains the real mean μ with probability at least 90%.

Figure 9 (b) plots the ratio $M_n/n^{1/4}$. It indicates that the constants in our analysis are low, that is, less than 1.6. Moreover, the method compares well with the theoretically best possible $O(\log n)$, which assumes both preprocessing and additional storage. Figure 9 (c) plots $M_n/\log_2 n$ and shows that, for the observed range of n , the number of visited tetrahedra stays well under $2.1 \log_2 n$.

7 Closing Remarks

Point location by walking through a triangulation is often used in the practice of geometric computing, and with excellent empirical results; in particular in 3D mesh

generation, see for example, [GH92, BL95, TG⁺96]. By simulating bucketing via random sampling, we enhanced the procedure to what we call the jump-and-march, and we were able to show that the new procedure has an expected running time of $O(n^{1/4})$ when applied to the 3D Delaunay triangulation of a uniformly distributed random point set of size n . Our own empirical tests verify this; in fact, the relaxed jump-and-walk procedure, which is even easier to implement, performs as good, or even better in terms of actual running time. Experiments with nonrandom data [M93] show similar results. Our work justifies a procedure which is often used in the “real world” and even suggests a simple yet effective improvement.

Our work raises several interesting questions both theoretical and empirical. 1) Our result shows that the method requires expected time close to $O(n^{1/(d+1)})$ for $d = 3$ dimensions. It would be very interesting to settle whether the method performs equally well for higher dimensions. Two things are required to adapt our proof to higher dimensions. First, in order to extend our Lemma 5 and bound the expected value of N_1 , one needs to define appropriately a “ d -dimensional spindle”: this geometrical object is composed of d balls whose centers are on a $d - 1$ -sphere positioned on the hyperplane between X_1 and y . A critical step is to provide a d -dimensional version of our Lemma 3. Second, one needs to provide a d -dimensional analog of the argument allowing to relate N_2 to N_1 . This step might be novel. Indeed, we believe that our results can be tightened and that 2): In all our results the $\log n / \log \log n$ terms can be removed. 3) Our results make explicit the perturbing influence of the boundary. The proof of Lemma 5 required to identify locally the boundary ∂C to a plane. It would be very interesting to quantify this in terms of the curvature of ∂C . 4) It would be extremely interesting to know how the method performs on non-Delaunay triangulations. 5) Finally it would be very interesting to construct an adaptive version of our algorithm in the context where many query points are considered. One possibility would be to incrementally improve the data-structure as more and more query points are located. This would allow to select the m points of phase (1) in a more optimal way than purely random, resulting in a point Y closer to the query point. The improved algorithm could be analyzed via amortized analysis.

References

- [AEI⁺85] T. Asano, M. Edahiro, H. Imai, M. Iri, and K. Murota. Practical use of bucketing techniques in computational geometry. In G. T. Toussaint, editor, *Computational Geometry*, pages 153–195. North-Holland, Amsterdam, Netherlands, 1985.
- [BEY91] M. Bern, D. Eppstein, and F. Yao. The expected extremes in a Delaunay triangulation. *International Journal of Computational Geometry & Applications*, 1:79–91, 1991.
- [BP91] J. Bonet and J. Péraire. An alternating digital tree (ADT) algorithm for 3D geometric searching and intersection problems. *International Journal for Numerical Methods in Engineering*, 31:1–17, 1991.
- [BL95] H. Borouchaki and S. H. Lo. Fast Delaunay triangulation in three dimensions. *Computer Methods in Applied Mechanics and Engineering*, 128(1-2):153–167, 1995.

- [BD95] P. Bose and L. Devroye. Intersections with random geometric objects. Manuscript, School of Computer Science, McGill University, 1995.
- [B81] A. Bowyer. Computing Dirichlet tessellations. *The Computer Journal*, 24:162–166, 1981.
- [B93] E. Brisson. Representing geometric structures in d dimensions: Topology and Order. *Discrete & Computational Geometry*, 9(4):387–426, 1993.
- [DL89] D. P. Dobkin and M. J. Laszlo. Primitives for the manipulation of three-dimensional subdivisions. *Algorithmica*, 4(1):3–32, 1989.
- [DMZ95] L. Devroye, E. P. Mücke, and B. Zhu. A note on point location in Delaunay triangulations of random points. 1995. Submitted to *Information Processing Letters*.
- [DL76] D. P. Dobkin and R. J. Lipton. Multidimensional searching problems. *SIAM Journal on Computing*, 5:181–186, 1976.
- [Dwy91] R. A. Dwyer. Higher-dimensional Voronoi diagrams in linear expected time. *Discrete & Computational Geometry*, 6:343–367, 1991.
- [E90] H. Edelsbrunner. An acyclicity theorem for cell complexes in d dimensions. *Combinatorica*, 10:251–260, 1990.
- [GH95] M. Garland, and P. S. Heckbert. Fast polygonal approximation of terrains and height fields. Technical Report CMU-CS-95-181. School of Computer Science, Carnegie Mellon University, 1995.
- [GH92] P. L. George and F. Hermeline. Delaunay’s mesh of a convex polyhedron in dimension d . Application to arbitrary polyhedra. *International Journal for Numerical Methods in Engineering*, 33(5):975–995, 1992.
- [GS78] P. J. Green and R. Sibson. Computing Dirichlet tessellations in the plane. *The Computer Journal*, 21:168–173, 1978.
- [GS85] L. J. Guibas and J. Stolfi. Primitives for the manipulation of general subdivisions and the computation of Voronoi diagrams. *Transactions on Graphics*, 4(2):74–123, 1985.
- [M93] E. P. Mücke. Shapes and Implementations in Three-Dimensional Geometry. Ph.D. thesis. Technical Report UIUCDCS-R-93-1836. Department of Computer Science, University of Illinois at Urbana-Champaign, Urbana, Illinois, 1993.
- [PS85] F. P. Preparata and M.I. Shamos. *Computational Geometry: An Introduction*. Springer-Verlag, 1985.
- [TG⁺96] H. Trease, D. George, C. Gable, J. Fowler, E. Linnbur, A. Kuprat and A. Khamayseh. *The X3D Grid Generation System*. In *Proceedings of the 5th International Conference on Numerical Grid Generation in Computational Field Simulations*, 239–244, 1996.